title: Testnets

Flashbots operates on Goerli and Sepolia, so that searchers can test Flashbots without risking real funds.

Bundle Relay URLS

Network	URL
Mainnet	https://relay.flashbots.net
Goerli	https://relay-goerli.flashbots.net
Sepolia	https://relay-sepolia.flashbots.net

Examples

Here's how to setup the Flashbots Bundle Provider in Ethers to use Goerli or Sepolia:

```js
const provider = new ethers.getDefaultProvider("goerli");
// uncomment the line below to use Sepolia
// const provider = new ethers.getDefaultProvider("sepolia");

const authSigner = new ethers.Wallet(
  '0x2000000000000000000000000000000000000000000000000000000000000000',
  provider
);

const flashbotsProvider = await flashbots.FlashbotsBundleProvider.create(
  provider,
  authSigner,
  // use "https://relay-sepolia.flashbots.net" for Sepolia
  "https://relay-goerli.flashbots.net",
  "goerli"
);
```

Sending bundles works the same as sending bundles on the mainnet. For example this will simulate a bundle and if it is successful then send a batch of 10:

```js
const wallet = new ethers.Wallet(SOME_PRIVATE_KEY);
const signedTransactions = await flashbotsProvider.signBundle([
  {
    signer: wallet,
    transaction: {
      to: "0xf1a54b075fb71768ac31b33fd7c61ad8f9f7dd18",
      gasPrice: 10,
      gasLimit: 21000,
      chainId: 5,
      value: 0,
    },
  },
  {
    signer: wallet,
    transaction: {
      to: "0xf1a54b075fb71768ac31b33fd7c61ad8f9f7dd18",
      gasPrice: 10,
      gasLimit: 21000,
      chainId: 5,
      value: 0,
    },
  },
]);

const blockNumber = await provider.getBlockNumber();

console.log(new Date());
const simulation = await flashbotsProvider.simulate(
  signedTransactions,
  blockNumber + 1
);
console.log(new Date());

// Using TypeScript discrimination
if ("error" in simulation) {
  console.log(`Simulation Error: ${simulation.error.message}`);
} else {
  console.log(`Simulation Success: ${blockNumber} ${JSON.stringify(
    simulation,
    null,
    2
  )}`);
}
console.log(signedTransactions);

for (var i = 1; i <= 10; i++) {
  const bundleSubmission = flashbotsProvider.sendRawBundle(
    signedTransactions,
    blockNumber + i
  );
  console.log("submitted for block # ", blockNumber + i);
}
console.log("bundles submitted");
```

The reason why we submit bundles for the next 10 blocks is because Flashbots only runs a small portion of the validators on Goerli. We are more likely to have a bundle included if we submit bundles for several blocks into the future.